Naval Research Laboratory
Washington, DC 20375-5320

NRL/MR/8140--98-8174

End-to-End Performance of Simplified ADF Backbone Infrastructure

JUNHO CHOI
ERIC SYDOW

Commands, Control, Communications, Computers and Intelligence Branch
Space Systems Development Department
Naval Center for Space Technology

CHRIS FUCHS

Mnemonics, Inc.
Alexandria, VA 22307

June 16, 1998

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave Blank*)

2. REPORT DATE

June 16, 1998

3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE

End-to-End Performance of Simplified ADF Backbone Infrastructure

5. FUNDING NUMBERS

6. AUTHOR(S)

Junho Choi, Eric Sydow, and Chris Fuchs*

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/8140--98-8174

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

SPARWAR, Code 9110
Naval Research Laboratory

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

*Mnemonics, Inc.
6304 Potomac Ave.
Alexandria, VA 22307

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

Main objective of this report is to develop models for the current ADF backbone infrastructure, to improve understanding of ADF network functionality as well as the capability of available network analysis tools. Preliminary results show that the smaller MTU experience less average ETE delay when errors exist in the network. Whenever an error occurs, a data packet has to be resent. This implies that the longer the packets to be resent, the more bandwidth required. Thus, smaller packets can be resent more effectively. The next report will emphasize extended scales of complexity of the current ADF network, and ATM-based backbone architectures.

14. SUBJECT TERMS

MSS	End-to-End delay	OPNET	DDF
ACK	SMI	FDDI	FIFU
Ethernet	MMF	ATM	CSMA/CD
Server/Client	LAN	FSM	TCP/IP
Packet Length	Protocols	MTU	

15. NUMBER OF PAGES

26

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18
298-102

CONTENTS

LIST OF FIGURE CAPTIONS

End-To-End Performance of Simplified ADF Backbone Infrastructure

1. Introduction

1.1. Aspects and Issues of Telecommunication Network Analysis

The field of modern computers and communications is being transformed approximately every two years by an order of magnitude in three core technologies: microelectronics, photonics, and software. In the field of communications, the circuit boards populated by VLSI (Very Large Scale Integration) chips perform the routing, protocol processing, storage and media access control needed to enable the low cost local networking among desktop computers and workstations distributed throughout large office buildings like the ADF (Aerospace Data Facilities in Denver, Colorado) facility`. Advances in the field of photonics have had significant impact on computer/telecommunications. Photonics technology provides the low loss, low dispersion, no electromagnetic interference, and extremely high bandwidth afforded by optical fiber and associated devices. When combined with narrow line-width single frequency semiconductor lasers and low noise optical receivers, it becomes possible for optical fiber links to support very high point-to-point digital data rates (e.g., up to terabytes (TB)) over very long unrepeated distances (e.g., single mode fiber (SMF)). In the software domain, powerful desktop workstations, file servers, object-oriented programming, distributed operating systems, distributed databases, and distributed computing have created the demand for telecommunication networks with features and capabilities surpassing those intended for a voice and voice grade data-dominated network traffic environment [1].

The architectures and fundamental nature of telecommunication networks are characterized by two distinct elements: switching and transmission systems. Transmission systems are responsible for the movement of properly formatted information on a point-to-point basis. Typically, information arising from a multitude of sources are multiplexed onto a common transmission medium which carries the information to a geographically remote destination where it is detected and demultiplexed into its constituent components. The widespread introduction of broadband service offerings to offices and each end user will pose fundamentally new challenges in network design. Multiplexing and transport techniques for both the interoffice (exchange) network and the subscriber loop must be geared for transmission capacity orders of magnitude greater than those typical in today's telecommunication environment. The large bandwidth-distance products available with single mode optical fiber make it the medium of choice for point-to-point transmission of broadband signals. However, in addition to providing high speed transmission capabilities, the network must also incorporate switching, routing, and multiplexing techniques that are appropriate for broadband services. Indeed, the efficient implementation of these signal processing functions can have an enormous impact on overall network cost.

The end points of the transmission system might be either remote terminals which gather and distribute information to end users, or switching nodes which terminate some multitude of transmission links. It is the responsibility of the switching system to route the information corresponding to each end user source to the correct outbound transmission system by forming the correct interconnection patterns among its input and output ports. The connections of each LAN (local area network) for example, might be carried to the switch by a common transmission system and then demultiplexed, switched, and remultiplexed for delivery, each to a destination in a different LAN cluster. The switch is

Manuscript approved May 14, 1998.

reconfigurable; that is, the interconnection patterns among its input and output ports can be changed in response to changing patterns among source-destination pairs.

The traditional boundary between switching and transmission systems is already becoming blurred, and in the broadband era, the functionality of these systems will be replaced by the transport network which has exclusive responsibility for the movement of information from geographically dispersed input ports to the correct output ports. The transport network should be layered according to function. This simplifies the design, development, and operation of the network and allows smooth network evolution. The introduction of the layered concept also makes it easy for each network layer to evolve independently of the other layers by capitalizing on the introduction of new technology specific to each layer [2]. The transport network may contain traditional point-to-point transmission systems and centralized switching systems, supplemented by (or entirely displaced by) multipoint transmission systems. Such multipoint systems allow information-bearing signals to enter and leave the transmission medium from multiple, geographically distributed access stations. Hardware-based intelligence located within each access station is capable of making local routing decisions (i.e., accept a segment of information, ignore that information, regenerate and relay the information, etc.), effectively creating a distributed switch. In the current ADF telecommunication infrastructure, large and centralized switching systems of this type are interconnected by point-to-point transmission systems such that a connection originating in one locality might be multiplexed, transported, and so forth, many times before arriving at its ultimate destination in some remote locality.

As communication applications have grown more sophisticated, a set of protocols or rules governing the flow of information and various quality checks were developed to enable reliable delivery of information in a relatively hostile network environment characterized by noisy, bit error-prone channels and congested network links, and switching nodes. Protocols were originally intended to ensure the integrity of delivered data, to control the flow of traffic over congested elements, and to monitor data lost, buffer overflow and other causes, to consume processing resources, and to constrain the suitable rate of information flow among terminating devices to the fastest that the protocol processor will permit. In addition to permitting the establishment and management of connections and keeping track of the multitude of connections which may be multiplexed through a common physical network port, protocols are also needed to enable reliable communications in a relatively hostile telecommunication environment characterized by noisy transmission and network congestion [3]. These impairments cause the appearance of random bit errors in delivered data, loss of segments of information from a delivered data packet or stream, and the misrouting of information to unintended receivers. To achieve the desired reliability, protocols check for errors in delivered information, search for missing segments, and control the rate of information flow to contain congestion at some acceptably low level.

A computer and communication network has both physical and functional (or logical) characteristics. The functional characteristics are defined by the communication mechanisms (network topologies, accessibility, packet sizing, etc) among the network elements. The physical characteristics are defined by geographical locations of transmission, switching, signal processing, and storage elements. There are many different performance measures for network traffic control and performance management. The complexity and difficulty of computer/telecommunication network analysis exists due to the inter-relationship among various parameters of both functional and physical network characteristics. For example, the performance of various routing algorithms may very well differ; some may have better performance than others on certain objective functions; others perform other objective functions well. The relative performance of algorithms also depends on specific network configurations. Major criteria for these routing algorithms may rely on the speed of response, the number of control packets transmitted, computational complexity, size of control packets, buffer space required, and looping and loop freedom. The most important performance parameters in network modeling and simulation analysis may be listed as:

*An end-to-end delay,
*Data throughput,
*Message precedence,
*Critical decision time,
*Transmission power,
*Routing prediction,
*Buffer size,
*Protocol efficiency,
*Blocking probability, and
*Cost efficiency.

On the other hand, the network management performance and services heavily emphasizes on the dependability of the communication network. The dependability of the computer and communication network consists of:

*Survivability/security,
*Reliability,
*Availability,
*Maintainability,
*Operability, and
*Connectivity/Accessibility.

The purpose of this network performance analysis is to provide an administrative as well as a technical tool to aid in the development of communication network performance improvements, to help decision makers understand current network capabilities better, to compare or consider alternate network designs, to retune other network parameters, to assess the effects of overhead, and to evaluate the performance of a given protocol and network components such as packet switches.

Successful adjustment to emerging technological advances and dynamic computer and communication network requirement changes begins with careful analysis of the new technological environment and mission needs: to understand what is different, what is the same, what is important and what is not. Based on the results of this analysis, time and resources can be focused on the areas that are most critical to success in this dynamic telecommunication and computer network environment of the future.

1.2. Objectives of Network Modeling and Simulation

The main objective of this task is to characterize, verify, and investigate the current and future ADF computer and communication network performance for network reliability, operability, connectivity and maintainability among/between four buildings and floors when there exist a need to realign current communication network and to change systems mission. These objectives may be achieved through several approaches. One approach is to determine the validity and capability of both current and planned future ADF network topologies to meet dynamic requirements of the mission. The other approach is to verify the capability of network management for ever increasing complexity and the dynamic environment concerning network connectivity, availability, security, survivability, and operational performance. In fact, the network is a system of systems, far more complex than the computer systems we have seen in the past. Interdisciplinary approaches will be required using advanced concepts of mathematics, traditional networking, circuits and systems engineering, computer architecture, operating systems, and applications. Furthermore the entire spectrum from theory and analysis, modeling and simulation, development and design engineering, performance evaluation, to implementation testing and measurement will be required. While traditional performance analysis remains important, the effort of this task will have to place more emphasis on new theories and models, particularly chaos theory and fractals, general systems theory, and game theory.

This interim report presents preliminary simulation results for those objectives from a simple network model to rather complex models. These complex models will be very analogous to the current ADF communication network (Phase I of the ADF collapsed backbone network) in order to improve understanding of network functionality as well as to learn the capability of available software tools (OPNET- Optimized Network Engineering Tool, and NetMaker) for the task. This interim report concentrates on the performance aspects of the basic network by emphasizing the following attributes:

*Average peak load,
*End-to-end delay,
*Effects of change in loading or packet spacing/length,
*Receiver buffer size, and
*Number of nodes or switches.

A brief overview of the OPNET tool is presented in section 2 to enhance the understanding of the simulation and analysis effort presented in this report. In section 3, a single FDDI (Fiber Distributed Data Interface) ring model is presented for a number of nodal variation, end-to-end delay, buffer size, packet size, and acknowledgment delay. In section 4, several scenarios have been presented based on different network configurations of LANs with routers, Ethernet Hubs and FDDI rings. Finally, section 5 presents some conclusions and future aspects of this task.

2. An Overview and Capabilities of OPNET

OPNET is a software tool capable of modeling and simulating computer and communications networks with detailed protocol modeling and system performance analysis. OPNET features include: graphical specification of models; a dynamic, event-scheduled simulation kernel; integrated data analysis tools; and hierarchical, object-oriented modeling. One merit of OPNET's hierarchical modeling structure is to accommodate special problems such as distributed algorithm development. Another merit is to deliver open systems methodology and an advanced graphical user interface known as the MIL 3 User Interface. The OPNET system is a set of tools which can be divided into three functional areas:

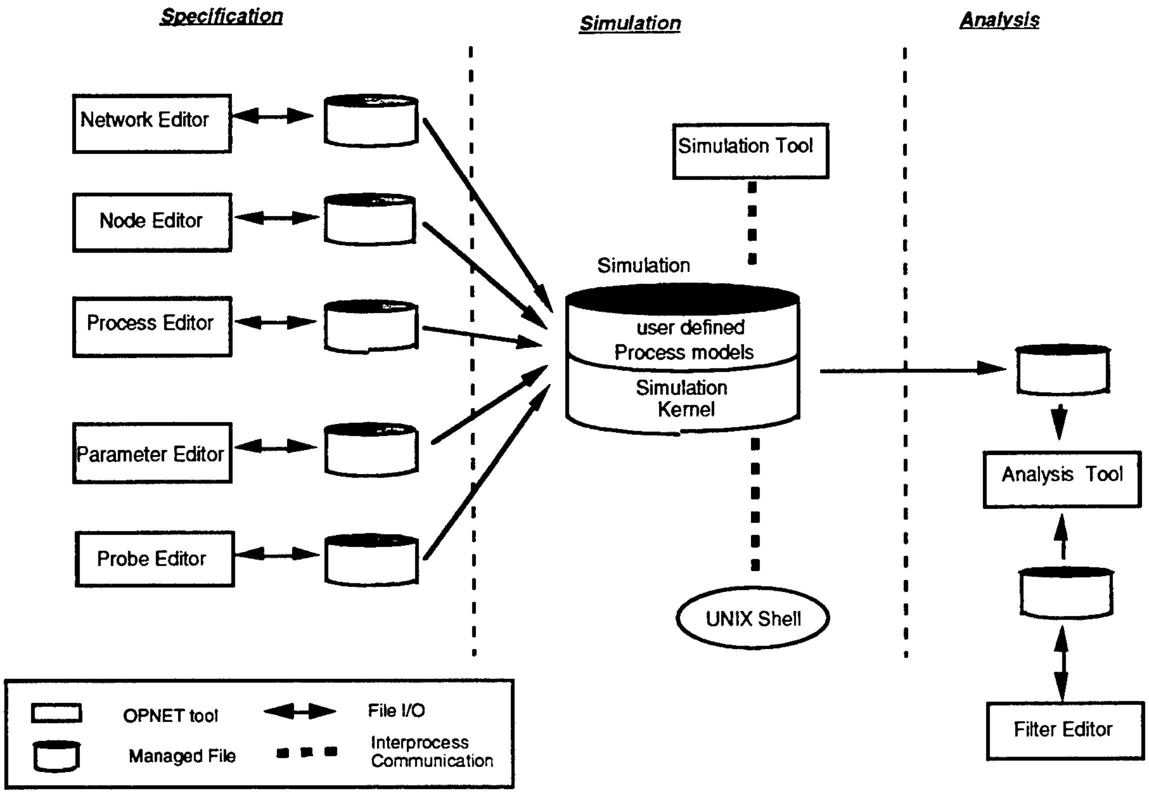
*Specification,
*Simulation, and
*Analysis

as shown in figure 1. The specification tool is further based on a four level modeling hierarchy consisting of network, node, process, and link level models. The simulation tool is used to prepare a configuration file for the network simulation and then to execute the simulation. The configuration file consists of probes of network variables that can be routed within the network for adaptive runtime behavior or saved into a simulation result file for later analysis. The kernel of the network simulation is a library of low-level functions which provide the simulation with the following services: an event scheduler, packet forwarding capability between nodes and internal to nodes, node modules such as queues and generators, data transmission models, link models, automated data capture, process model error checking, and simulation tracing. Finally, the analysis tool is used to analyze simulation result data that has been saved using probes set in the simulation tool. Data vectors can be plotted with a variety of graph types, and they can be numerically processed using filters that have been specified using the filter editor. The filter editor then specifies filters that will process simulation result data in the analysis tool. Filters are interconnected graphs of numeric processing elements such as summers, multipliers, differentiators, integrators, averagers, correlators, etc.

At the network level, nodes are either for communication (i.e., a user specified node model) or jamming (built into the system). Nodes can either be fixed in location or mobile on the ground or in the air; initial locations and trajectories for mobile nodes are specified. Point-to-point data transmission links between nodes are specified prior to simulation, while broadcast radio frequency (RF) links are dynamically computed during the simulation. Failures and recoveries of nodes as a probabilistic function of time can be specified at this level. The node level consists of six types of modules which operate on streams of packets and which can be interconnected to form complex node models. The processor module executes process models. The queue module serves as a FIFO buffer for packets. The generator module stochastically produces packet according to user-specified probability density functions (PDF). Transmitter and receiver modules model the transmission path between nodes. And the antenna module models the directional gain of transmitted and received packets. Process models can be represented either as finite state machines (FSM) or C language procedures. FSM's are specified using a graphical editor which captures the diagram of the state machine plus the logic of the transitions and states. C-based process models can be prepared and debugged with the popular UNIX programming tools. Both types of process model make use of a library of support procedures which allow access to packets and network variables. Finally the link level models incorporate custom or user-specific RF link models within OPNET simulations. The communication link between each transceiver pair is modeled as a pipeline which provides flexibility in specifying the transmission media between any two nodes. Link models are linked into the simulation, and are specified in C language.

The design methodology for simulations using OPNET tool is bottom-up in that the user first creates process models, then constructs node models which execute the processes, and finally constructs network models that are populated with node models communicating through links defined by the link level models. Top-down methodologies can be supported by stubbing out references to the lower levels until they are ready. The process models in the figure 1 can be represented either as FSM or C procedures. FSM's are specified using a graphical editor which captures the diagram of the state machine plus the logic of the transitions and states. C-based process models can be prepared and debugged with the popular UNIX programming tools. Both types of process models make use of a library of support procedures which allows access to packets and network variables. The link level models incorporate custom or user specific RF link models within OPNET simulations. The communication link between each transceiver pair is modeled as a pipeline which provides flexibility in specifying the transmission media between any two nodes. Link models are specified in C and are linked into the simulation. The node level models consist of six types of modules which operate on streams of packets and which can be interconnected to form complex node models. The processor module executes process models. The queue module serves as a FIFO (first in first out) buffer for packets. The generator module stochastically produces packets according to user-specified PDF. Transmitter and receiver modules model the

Figure 1: OPNET System Structure



transmission path between nodes, and the antenna module models the directional gain of transmitted and received packets. At the network level, these nodes are either used for communication (i.e., a user-specified node model) or jamming (built into the system). Nodes can either be fixed in location or mobile on the ground or in the air; initial locations and trajectories for mobile nodes are specified. Point-to-point data transmission links between nodes are specified prior to simulation, while broadcast RF links are dynamically computed during the simulation. Failures and recoveries of nodes as a probabilistic function of time can be specified at this level. A fifth level of modeling is also available for specifying certain parameters such as antenna patterns, PDFs, signal-to-noise ratio (SNR), bit error rate (BER) curves, time profiles of failures and recoveries, and packet formats. These models are referenced by the relevant modules within nodes.

3. A Case Study of a Single FDDI Ring Model

LANs have evolved based on the way information changes hands in the local area network environment. The office environment contains many intelligent machines such as computer systems, printers, plotters, terminals, and fax. For groups of people to function efficiently they must be able to exchange information effectively. The medium for obtaining this effective transfer of information, within the group and externally, is the LAN. The major product goals for optimal local area communication are bandwidth, reliability, maintainability, low cost, and simplicity. A brief introduction to Ethernet and the FDDI LANs, the two main mediums used for networking, and their appropriate interconnection devices are presented here.

FDDI is a set of ANSI (American National Standard Institute) standards for a fiber-optic, token ring local area data transport. The interface was conceived and developed by the ANSI X3T9.5 subcommittee under the Computer and Business Equipment Manufacturers Association, and approved for international use by the International Standards Organization (ISO). The set of standards defines a LAN topology consisting of a fiber optic token-ring based network with counter-rotating rings that supports a data transfer rate of 100 Mbps. Initially based on the use of fiber optics technology and advanced product design, FDDI networks can transfer increased amounts of data much faster and over longer distances than conventional Ethernet LANs. FDDI was designed to operate over optical fiber at up to 100 Mbps (megabits per second [sec]), and was conceived as a backbone for other, slower LANs. FDDI was also designed to be more robust than copper-based token rings. Each node was given active clock recovery and regeneration, could transmit multiple frames on a single token, and was equipped with extensive specially developed LAN management capabilities [5]. FDDI uses a time-token protocol (IEEE 802.5) to coordinate station access to the network. There can be multiple frames on the network which is configured as a logical dual ring or a dual ring of trees. The ring can cover a very wide geographical area. The maximum station separation is 2 kilometers (km) for MMF (multi-mode fiber) and greater than 20 kilometers for SMF (single-mode fiber). Up to 500 nodes may connect one ring in star, ring, or hierarchical star topologies. FDDI poses a solution to the growing congestion of older networks brought about by the increased use of high performance and fast desktop workstations, servers, and graphical interfaces.

Network congestion has many causes, such as an increase in the number of users, the increased power of modern desktop computers, and an expansion of long distance business applications. FDDI has been developed to meet those requirements and needs for networks to support graphics-intensive applications, and the increase in multimedia applications. The power and versatility of the FDDI network allows it to be currently configured in many ways (e.g., inter-directorate network which links various directorate LANs within ADF facilities): as a high speed backbone, high speed work group, and host-to-host/host-to-peripheral configurations [6]. FDDI networks offer four typical configurations: standalone, dual ring, tree of concentrators, and dual ring of trees. FDDI is an essential element of the current ADF computer and communication LANs and WANs (wide area networks) as well as of planned future ADF network configurations. This preliminary report presents performance measures of a standalone FDDI ring model only. Simulation results of node variation, acknowledgment delay, and effects of packet size are presented in this section.

Ethernet (IEEE 802.3 Carrier Sense Multiple Access with Collison Detection-CSMA/CD) provides the services of the lowest two layers of the ISO model: physical and data link. The physical layer characteristics for Ethernet are: 10 Mbps data rate, 2.8 km maximum station separation, 1024 maximum number of stations, twisted pair, coaxial cable optical fiber medium, bus logical topology, 1518 Bytes (B) of maximum frame size, single frames on LAN, fully distributed peer protocol link control procedure with statistical contention resolution, and variable frame size message protocol. Ethernet is a carrier sense protocol, i.e., all stations monitor the cable during their transmission, terminating transmission immediately if a collision is detected. When an Ethernet station wishes to transmit a packet, a carrier sense is performed forcing the station to defer if any transmission is in progress. If no transmission is detected, then the sender can transmit after an appropriate delay. It is possible that two, or more, stations will sense the channel idle at the same time and begin transmitting. This has the possibility of producing a collision. When a collision is detected, the station will stop transmitting and will reschedule a transmission at a later time. Retransmission time is random and is selected using a binary exponential backoff algorithm.

Figure 2 describes a single FDDI ring topology with the relationship between ISO and ANSI X3T9.5 layered architecture for five workstations and a single server. FDDI ring architecture is an essential network topology implemented in ADF facilities and it will be easily accessible to asynchronous transfer mode (ATM) switching technology in phase III plan. A brief explanation of modeling and simulation assumption is provided for reference to the following simulation result with graphical analysis and ISO layer interrelation.

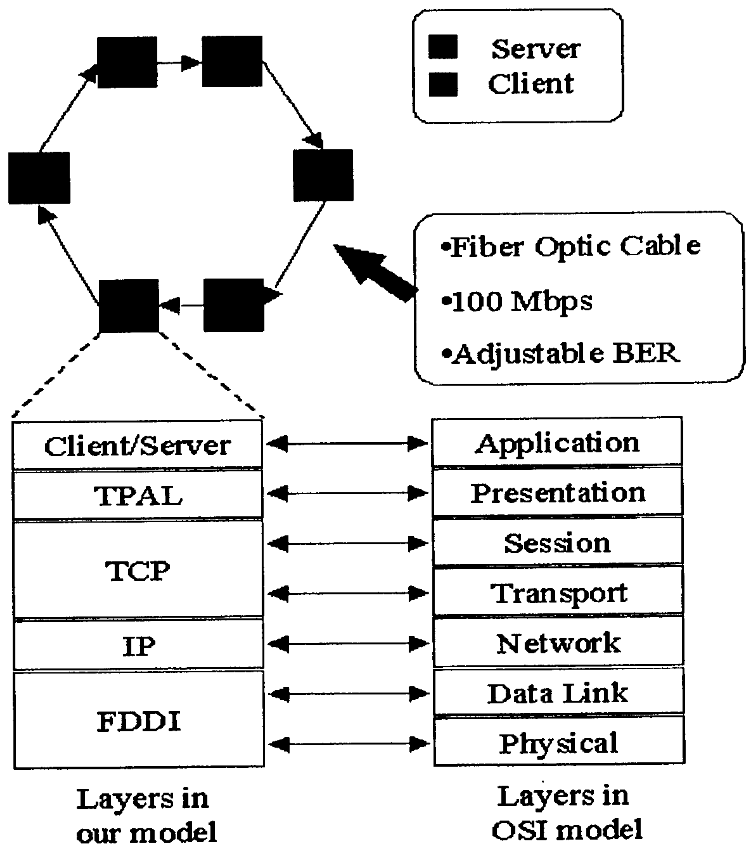


Figure 2: A Single FDDI Ring Architecture for 5 workstations and one server with layers

The following assumptions are made for this FDDI ring topology simulation as follows:

FDDI MAC(media access control) layer parameters -		constant
IP (internet protocol) parameters -		constant
TCP (transmission control protocol) parameters -		constant
TCP buffer size in bytes -	32 K, 65 K, 1.2 M, 5.0 M, 10 M, 50 M	
Retransmission time out (RTO) parameters -	initial value:	0.25 secs
	maximum value:	1.0 secs
	minimum value:	240 secs
Traffic pattern -	one way	
Single Ethernet source		
ACK delay in seconds -	0.0, 0.001, 0.01, 0.1, 0.2, 0.4	

Round Trip Time (RTT) -	deviation coefficient:	4 secs
	gain:	0.25
Persistent time out -	1.0 sec	
Throughput -	80 Mbps	
Packet generation -	Poisson distribution	
Packet lengths in bytes -	500, 1000, 2000, 4000, 8000, 16 K, 32 K, 64 K	

The procedure for generating packets is presented here for clarification. There exists a need to specify an average rate for packet generation so that the total amount of traffic for each of the five workstations equals 16 Mbps. This average packet rate is a function of the specified packet size. To compute the average number of packets per hour, the following formula is adopted as:

$$P = k^*(L)^{-1}$$

where P is the number of packets per hour, L the length of packets, and k is a constant given by

$$k = [(3600 \text{ secs/hour})^*(16 \text{ Mbps/workstation})/8 \text{ bits/byte}]$$

Two parameters, throughput and end-to-end delay, are analyzed for this simulation. Throughput is the measure of whether data are successfully transmitted through the network or not. If the network's throughput matches the traffic load, the data are transmitted through the network. On the hand, if the throughput rate is less than the data traffic load, the network fails. An end-to-end (ETE) delay is the time it takes for a packet to be sent through the network. OPNET measures this quantity by associating a timer with each packet it sends through the network. If the ETE delay is measured in the TCP layer from one node to another node, then OPNET starts this timer when the packet leaves the TCP layer of the first node, and it ends the timer when the packet reaches the TCP layer of the destination node. The total time is the ETE delay. Since the delay is one of the most important measures of performance for the network's quality of service (QoS), excessive delays are not suitable for real-time applications, require a large buffer, and annoy users.

Figure 3 shows the effects of changing the amount of nodes on one ring. This graph shows the cumulative probability of end-to-end delay for rings with different amounts of nodes. As we increase the amount of nodes on the ring the cumulative probability shifts to the right, meaning it is more likely for a packet to experience higher delay when there are more nodes on the ring. The amount of delay increases approximately linearly with the number of nodes.

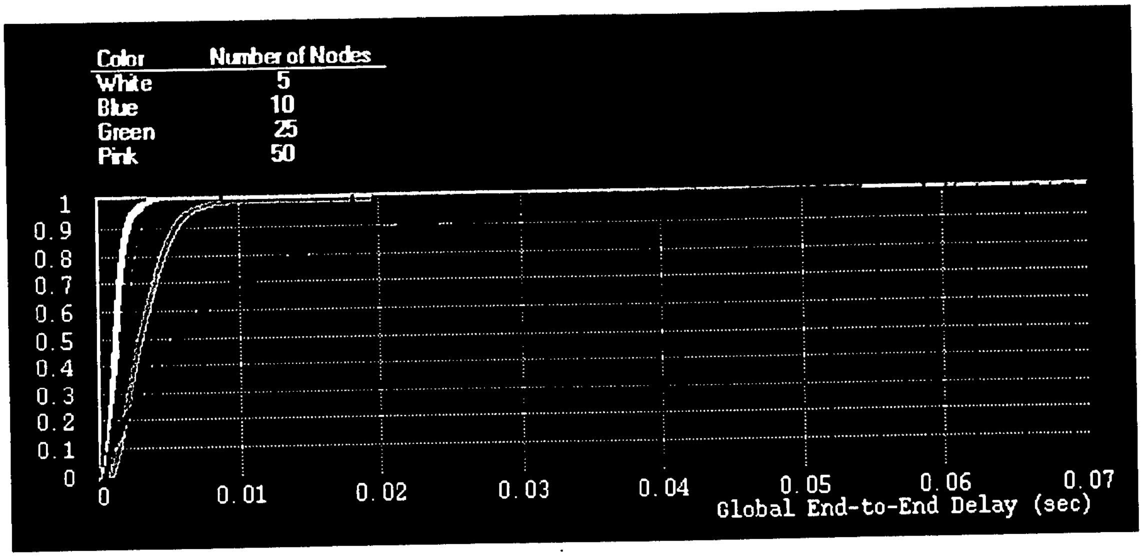


Figure 3. Delay increases as nodes are added to the ring

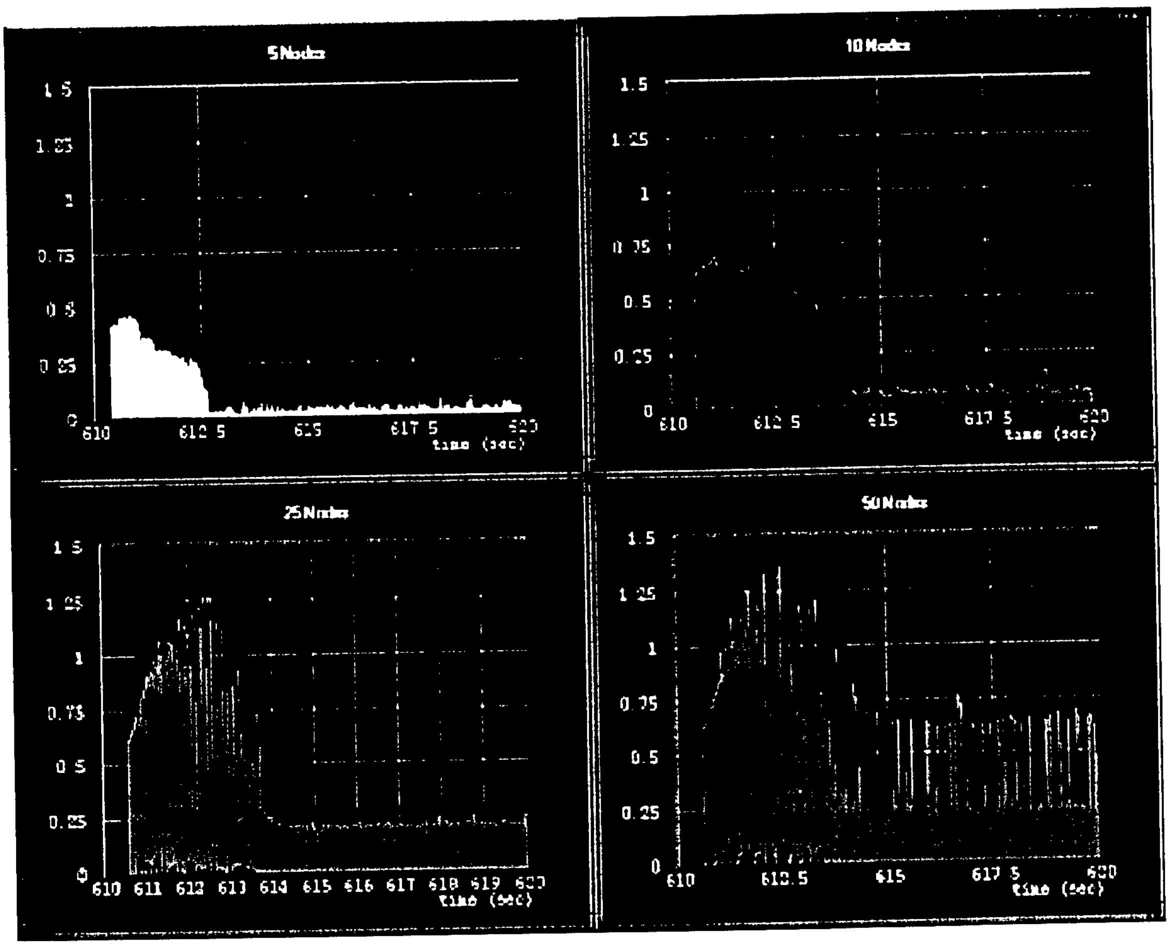


Figure 4. End-to-End Delay for nodes of 5 (white), 10 (blue), 25 (green), 50 (red)

Figure 4 shows the end-to-end (ETE) delays during a 10 second simulation for four different numbers of nodes- 5 (white), 10 (blue), 25 nodes (green), 50 nodes (red). The smaller the number of nodes is , the smaller the ETE delay is. When the number of nodes are larger than 25, the ETE delay gets larger and inconsistent. However, regardless of the number of nodes on the ring the general trends stay the same. Therefore regardless of the number of nodes the general results obtained by our simulations will hold. Note here that when choosing parameters at the higher layers of the ISO reference model, the network configuration works best when tailored to the lower layers regardless of the data link service. The system network will also work best when packets at the higher layers are smaller than the maximum segment size (MSS) regardless of MSS used by the lower layers. MSS is chosen to best utilize data transmission scheme when the standards invariably chooses an MSS [7]. For example, the creators of the Ethernet chose to best utilize data to optimize a trade-off between overhead and collisions. Since network link technologies have been standardized and a hardware has been designed accordingly, MSS of the given network is decided when a new equipment is purchased. The following step is thus obviously to choose parameters of the upper layer network protocols to maximize capabilities of available data link technologies.

4. Analysis of Generic ADF LAN Architectures and Issues

An early version of the ADF network architecture configuration is shown in figure 5 for many different cross program internetworking relations. In this configuration, an end-to-end delay has been investigated by assuming standard one way TCP/IP file transfer for all cases based on receive buffer size, packet length, acknowledgment delay. Six different cases are examined for different routing and switching/server combinations.

4.1. A Single Source to Single Server Case with FDDI server

The ETE delays through a Cisco router and a filter router for source #1 to server #1 in the figure 5 is examined with the assumption of:

Receive buffer size -	65 KBytes (KB)
ACK delay -	0.0 seconds (secs)
Ethernet source -	one
Network path -	source #1 to server #1
Traffic pattern -	one way traffic
Server type -	FDDI server
Client/server source -	4 Mbps
Packet lengths in Bytes -	512, 1000, 1300, 1400, 1500, 2000

Even though the maximum transmission unit (MTU) size of Ethernet is 1500 B, only 1460 B are available for raw data due to 40 B allocated for TCP/IP headers. If the data packet's size is larger than 1460 B, it will be segmented into more than one Ethernet packet. This causes either more overhead or more delay. If data packet is segmented, there are a large number of packets. This implies that it needs more header information, i.e., more overhead. On the other hand, if first segmented packet arrives at the receiving end, it will wait for other packets from original data packet before it can be reassembled and sent to the server's application layer. This additional segmentation causes an additional increase of the ETE delay variation, i.e., more delay. Figure 6 demonstrates the ETE delay for six different

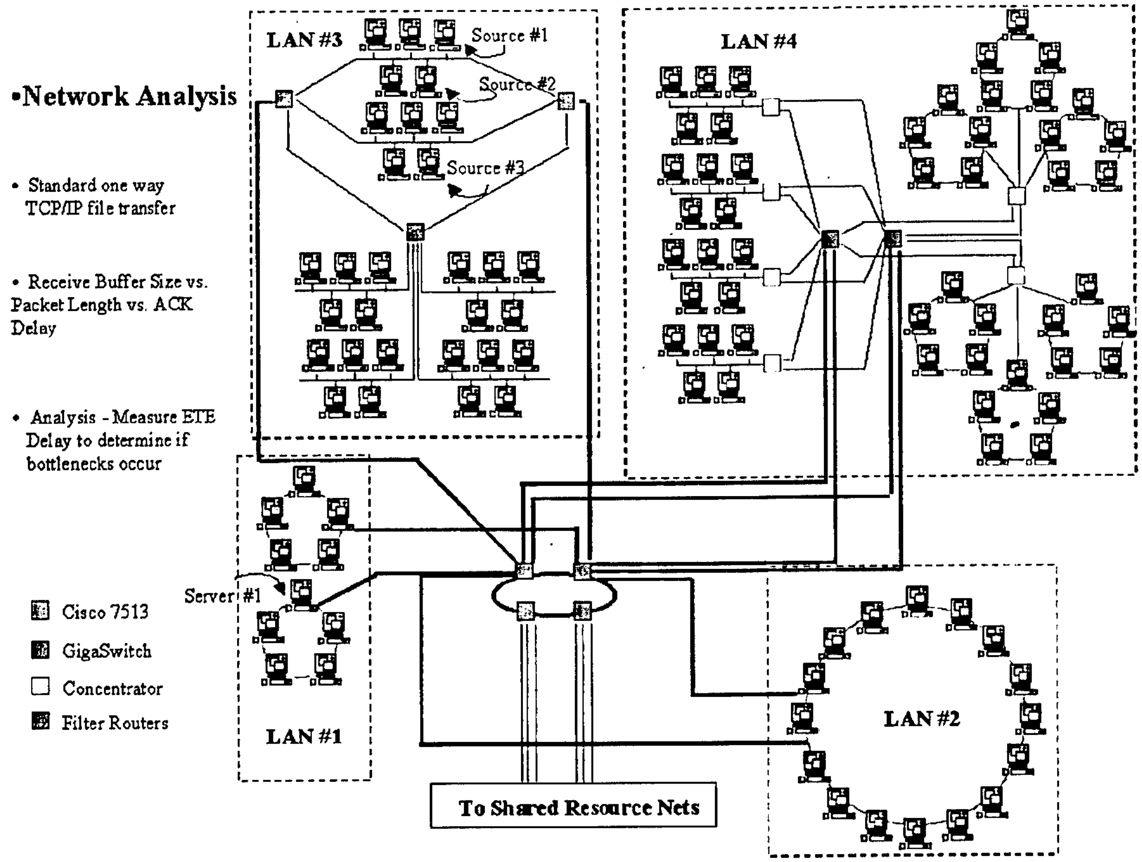


Figure 5: Current Generic Consolidated ADF Network Architecture

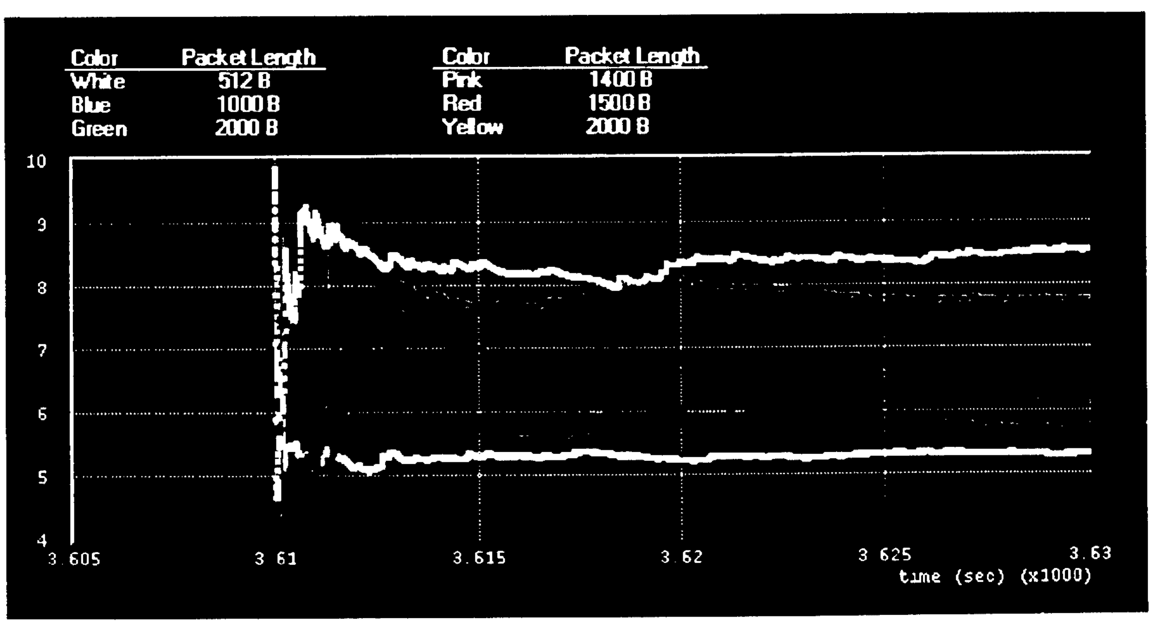


Figure 6. An Average End-to-End Delay for Several Data Packet Lengths.

packet lengths for the period of 20 secs. One can see that smaller packets cause less ETE delays. Note the noticeable increase in ETE delay (about 2 millisecond [ms]) from 1400 B case to the 1500 B case even though the increase in packet size is only 100 B.

Figure 7 shows the same case except with two receive buffer sizes (65 KB and 1.2 MB) and four packet lengths-512 B, 1000 B, 2000 B, 4000 B. The receivers specify buffer sizes, the transmitters thus acknowledge how much data they are allowed to send. Transmitters cannot send more than this specified amount of data. The larger the size of the receiver buffers, the more data the transmitters can send each time. As shown in the figure 7, the ETE has little effect on increasing buffer size. For example, the ETE delay is exactly same for the cases of 512 B and 1000 B packet sizes. The 65 KB receive buffer is more than enough to handle 4 Mbps data rate of the transmitter. Similar results were found for the cases of 10 MB, 50 MB, and 100 MB buffer sizes. Notice that the increase in buffer sizes has more effect on the ETE delay when the source data rate is increased. When the transmitter sends a large amount of data, the transmitter must wait for the acknowledgments (ACKs) before it sends another large set of data. This process increases the variation of ETE delay, which causes the average ETE delay value to increase.

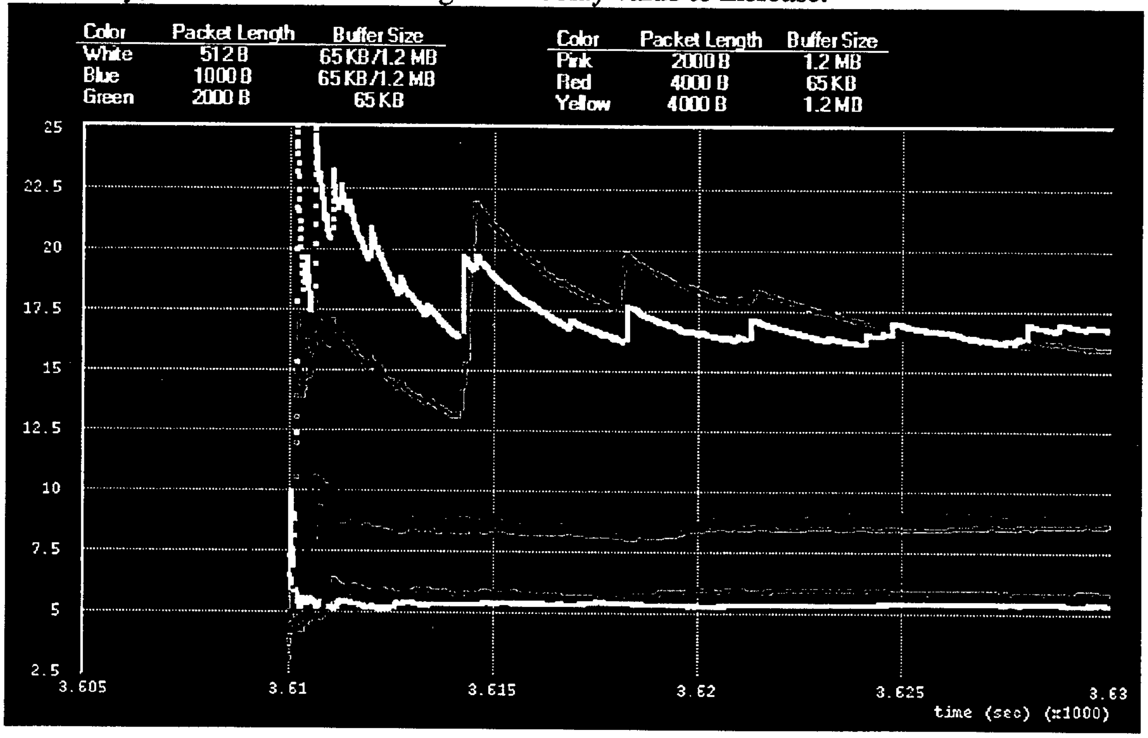


Figure 7. Average ETE Delay for Several Data Packet Lengths and Buffer Sizes

Figure 8 examines the same case as in figure 7 except for an additional 0.01 sec of ACK delay. It is possible that there may be traffic heading back which an ACK packet can piggyback on so that the number of packets and the amount of traffic on the network decreases. By changing the ACK delay, one can determine how long the receiver waits for returning traffic before it will give up and send a single ACK packet. The ACK will be delayed every time since we assume no returning traffic. Therefore, each ACK packet will acknowledge more source traffic at a time, so there will be fewer ACK packets. But this benefit cannot offset the cost of having ACK packets delayed in the first place. When the ACK delay is increased, the average ETE delay is increased as shown in the figure 8, and similar results, although not shown here, were found for ACK delays of 0.001 secs, 0.2 secs, 0.4 secs.

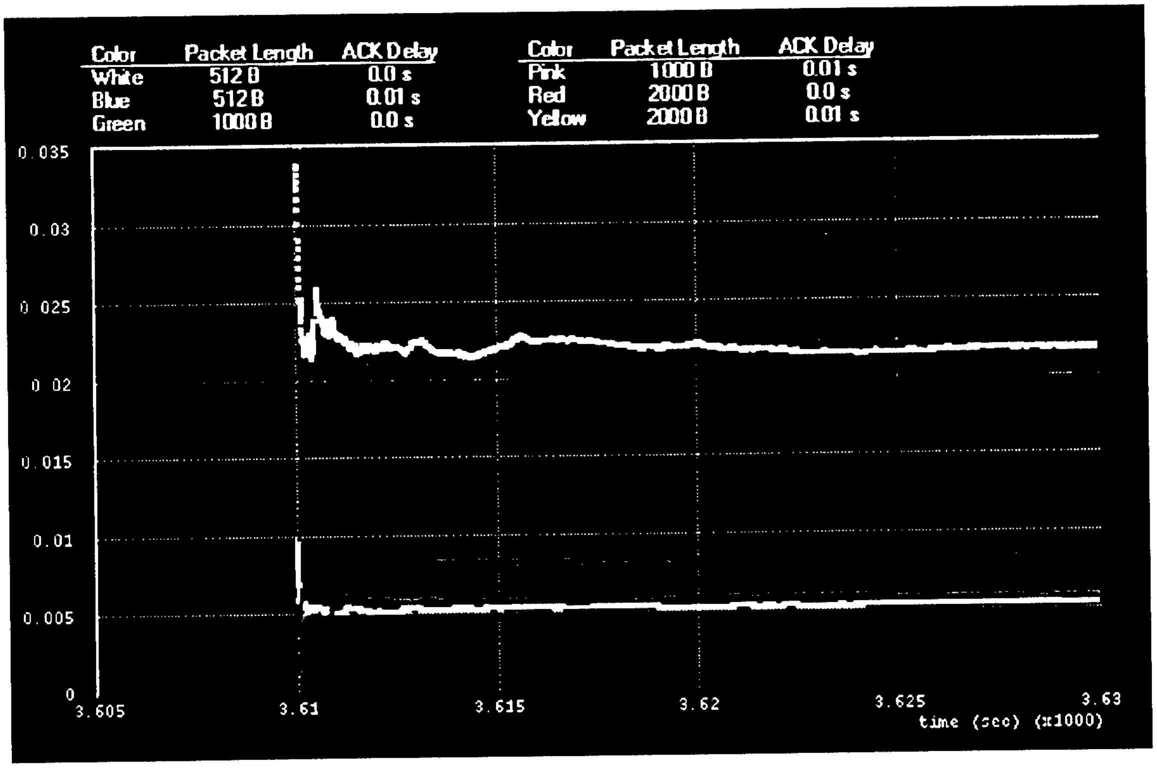


Figure 8. Average End-to-End Delay for Several Data Packet Lengths and ACK Delay

In summary for case 1, a significant increase in ETE delay is noted when fragmentation occurs around 1500 B packet lengths because there is no bandwidth constraints due to an application of only 4 Mbps of source data rate. When packets size is appropriate for default buffer size of 65 KB, the increase of the buffer size does not seriously affect since the receiver buffer size is large enough to handle 4 Mbps source rate in this particular circumstance.

4.2 Two Sources within the Same LAN to a Single FDDI Server

The number of sources is increased to two sources within the same Ethernet LAN from sources #1 and #2 to server #1 in figure 5. The average ETE delay for different buffer sizes is examined in this case with assumptions made as follows:

Receiver buffer size - 65 KB and 1.2 MB
ACK delay - 0.0 secs
Two Ethernet sources from the same subnetwork
Network path - sources #1 and #2 to server #1 in figure 5
Traffic pattern - one way traffic
Server type - FDDI
Clients/server source - 4 Mbps from each client
Packet lengths - 1000 B, 8000 B

Figure 9 demonstrates the ETE delay for two different buffer sizes. When the buffer is 65 KB, not enough traffic flows through the network, and the data back up at the transmitter. In order to resolve this traffic slow down, buffer sizes are increased to 1.2 MB. The traffic overload was not relieved since the two sources compete with each other for the same bandwidth. When more bandwidth is available to one source during the simulation, its ETE delay gets lower. This trend reverses itself when the other source has access to more bandwidth. When the buffer size increases to 1.2 MB (green and pink graph in figure 9), the clumping effect occurs. Because the transmitter sends data whenever 1.2 MB of data dumps into the network, the transmitter must wait until this data is fully received and acknowledged before it can send more data.

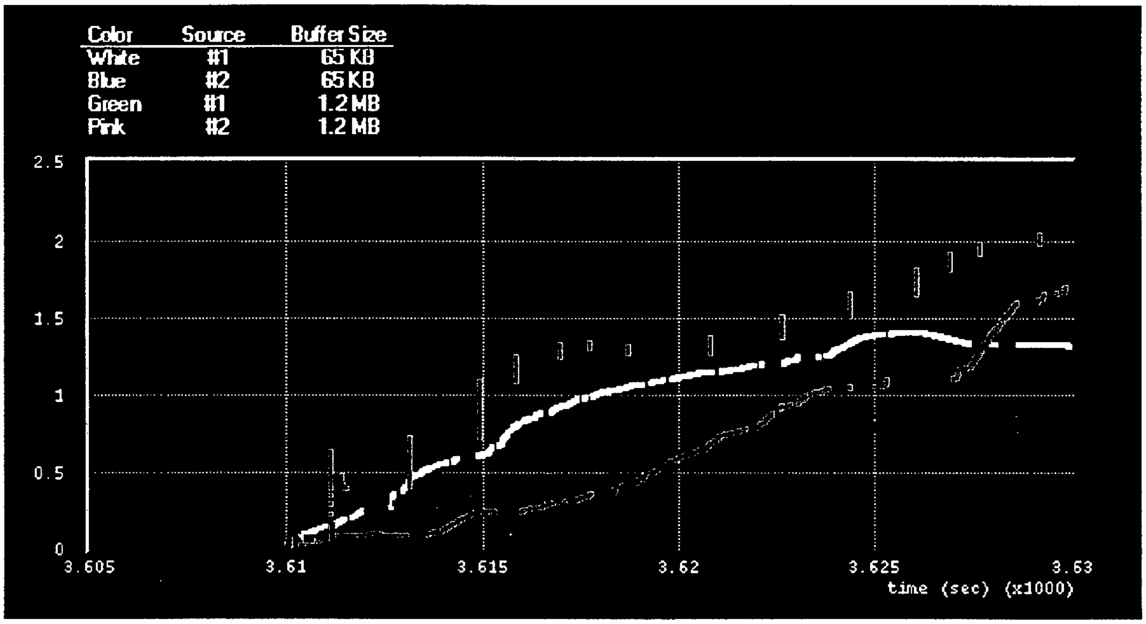


Figure 9. Average End-to-End Delay for Different Receive Buffer Sizes

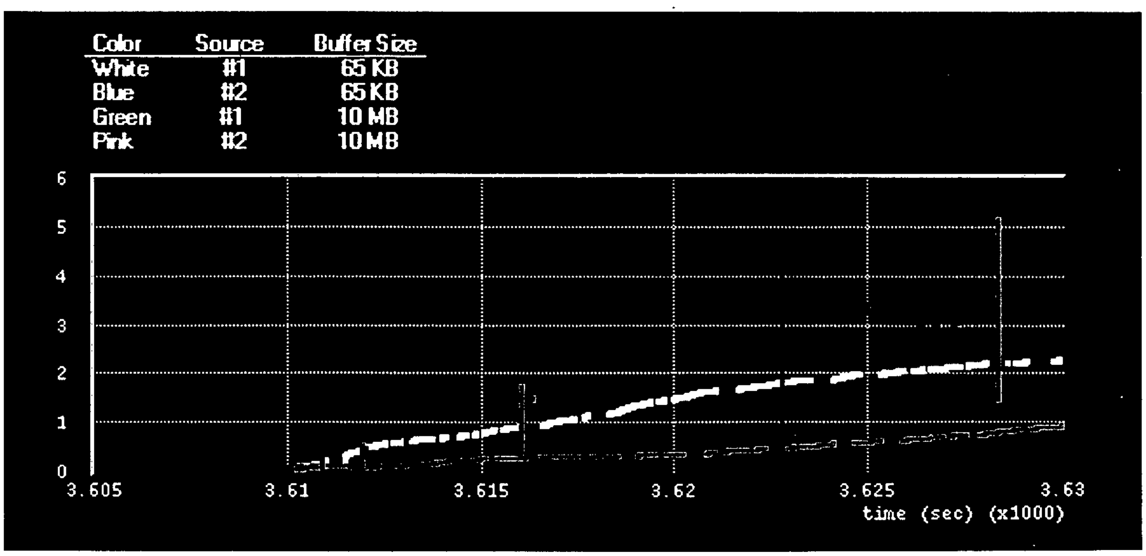


Figure 10. Average End-to-End Delay for Different Receive Buffer Sizes

Figure 10 presents a similar case to that in the figure 9 except for an increase in buffer size to 10 MB and a change of packet length to 1000 B. When the buffer size is increased to 10 MB, the clumping effect is much larger and ETE delay is higher. This implies that the increase of buffer size alone does not solve the network traffic congestion unless other parameters are coordinated with it.

4.3. A Single Source to A Single Ethernet Server

This case is similar to the case of 4.1 except that an Ethernet server is used instead of an FDDI server. Specifications are as follows:

Receive buffer size -	65 KB
ACK delay -	0.0 sec
Single Ethernet source	
Network path -	source #1 to server #1 in figure 5
Traffic pattern -	one way traffic
Server Type -	Ethernet
Clients/server source -	4 Mbps
Packet lengths in Bytes -	512, 1000, 2000, 4000

Simulation results indicate that the ETE delay increases significantly for all packet sizes due to the extra Ethernet links. For this case all network links are constrained to the 10 Mbps Ethernet capacity. When we compare these results with the case of 4.1, the ETE delay for Ethernet is 8 to 10 ms higher than for FDDI as shown in the figure 11.

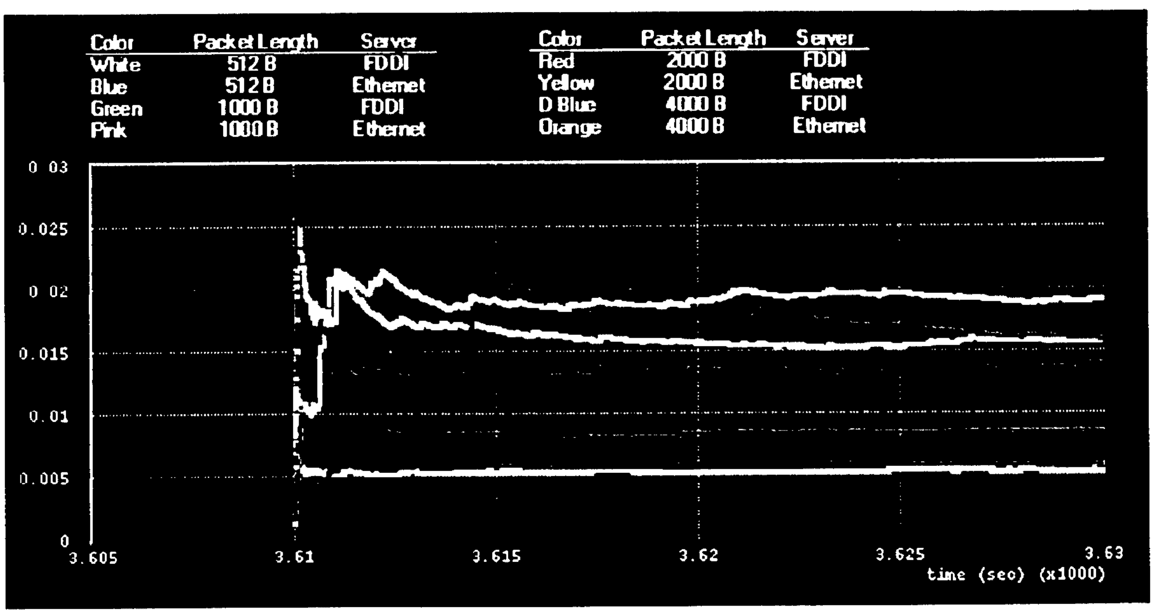


Figure 11. Ethernet Client to Ethernet or FDDI Server

4.4. A Single Source to A Single FDDI Server with Higher Source Rates

This case is exactly the same as the case in 4.1 except the source data rates increase to 8 Mbps. Details of simulation specifications are given as follows:

Receive buffer size -	65 KB and 1.2 MB
ACK delay -	0.0, 0.001 sec
Single Ethernet source	
Network path -	source #1 to server #1 in figure 5
Traffic pattern -	one way
Server type -	FDDI
Clients/server source -	8 Mbps
Packet lengths in Bytes -	1000, 2000, 4000

Simulation results reveal that a receive buffer size of 65 KB is not adequate and the transmitter needs to send more than 65 KB each time. After the buffer size is increased from 65 KB to 1.2 MB, the ETE delay stabilizes. Figure 12 shows that the ETE delay for packets with a length of 2000 B and a 65 KB receive buffer is lower than that for packet with a length of 4000 B, while the ETE delay for packet lengths of 2000 B and 1.2 MB receive buffer is higher than that of the packet length 4000 B. This contradicts with the previous results, where ETE delay for the smaller packet length is always lower than that of the larger packet length. One should realize that when the network operates near its bandwidth limit, it is very sensitive to the source data rate. During these simulations, a pseudo random number generator is used to generate the 8 Mbps source data for network traffic. The source data rate is not always guaranteed to be exactly 8 Mbps; it is usually a little bit more or less than 8 Mbps. This causes the ETE delay to be much larger even for rates slightly more than 8 Mbps. Note that the change in buffer size has affected more than the change in packet lengths.

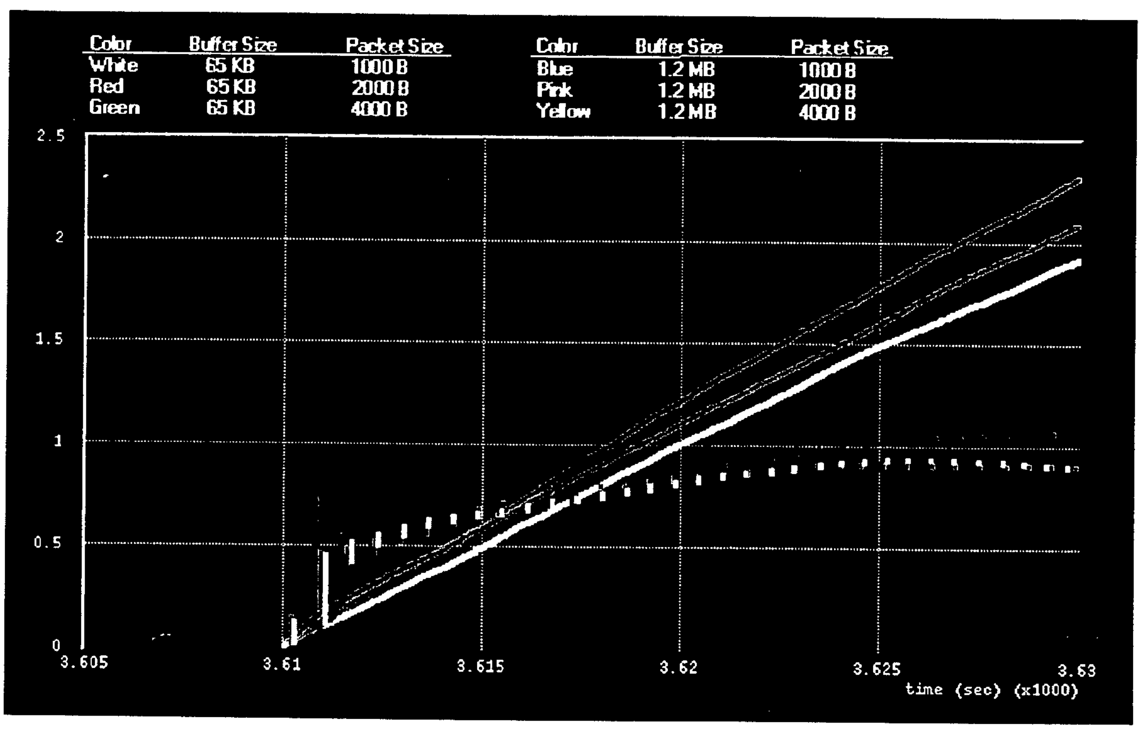


Figure 12. 80 % Traffic Load for Different Buffer Sizes

Figure 13 presents four different cases with a packet length of 1000 B for three different receive buffer sizes of 65 KB, 1.2 MB, and 10 MB with additional ACK delay of

0.001 secs. Notice that the network starts to stabilize when the receive buffer size changes from 65 KB to 1.2 MB. This results in a decrease of the ETE delay. Also note that when the receive buffer size increases to 10 MB, average ETE delay increases dramatically. Note here that the minimum values in ETE for the 10 MB receive buffer size case are equal to those for the 1.2 MB case. However, the maximum values for the 10 MB case are dramatically higher. The reason is that as the buffer size increases, a large amount of data is transmitted each time, and the transmitter has to wait longer between transmission of data.

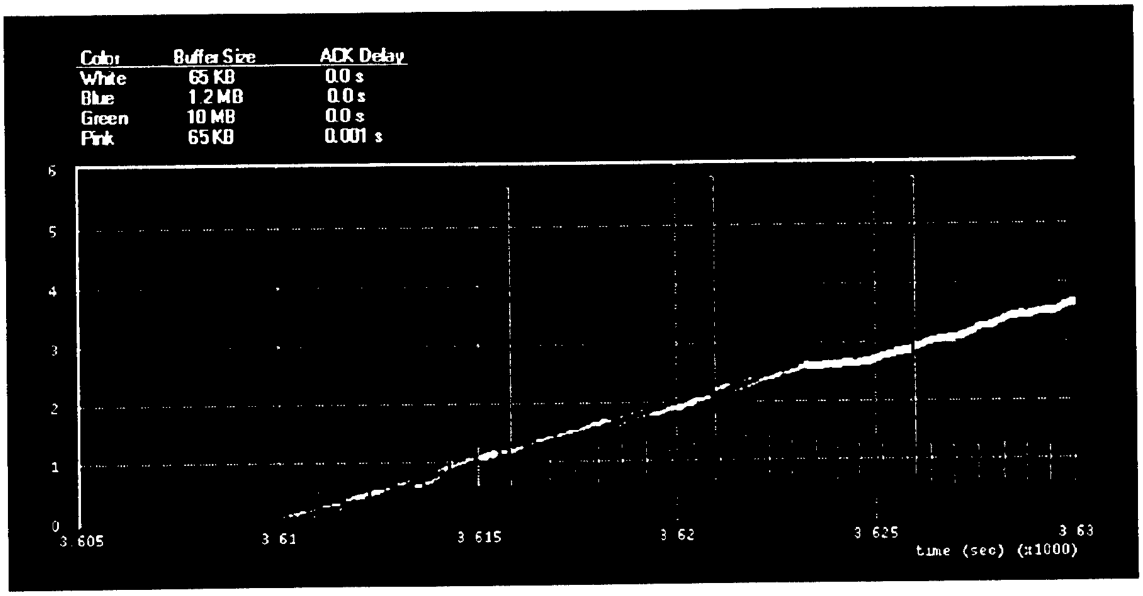


Figure 13. 80 % Traffic Load for Different Buffer Sizes and ACK Delays

4.5 A Single Source to A Single FDDI Server with Bit Error Rate

The relationship between ETE delay and bit error rate (BER) is examined in this case with the following network related attributes as:

Receive buffer size -	65 KB
ACK delay -	0.0 sec
Single Ethernet source	
Network path -	source #1 to server #1
Traffic pattern -	one way
Server type -	FDDI
Clients/server source -	4 Mbps
Packet lengths in Bytes -	512, 4000, 8000
BER rates -	1E-7

Figure 14 demonstrates that cases for the error in the network and packets transmitted three times with different packet lengths, and shows how the BER affects three different packet lengths and what happens to the case of a single error causing a retransmission. Simulation results show cases for all three packet lengths performs similarly in that each case waits approximately 0.5 sec before sending a retransmission. When the retransmitted packet is acknowledged, the network catches up quickly within about 0.4 secs. The ETE delay is not greatly different for the three different packet lengths due to small source rates, 4 Mbps, which is much smaller than enough bandwidth to handle retransmission for the Ethernet bandwidth of 10 Mbps.

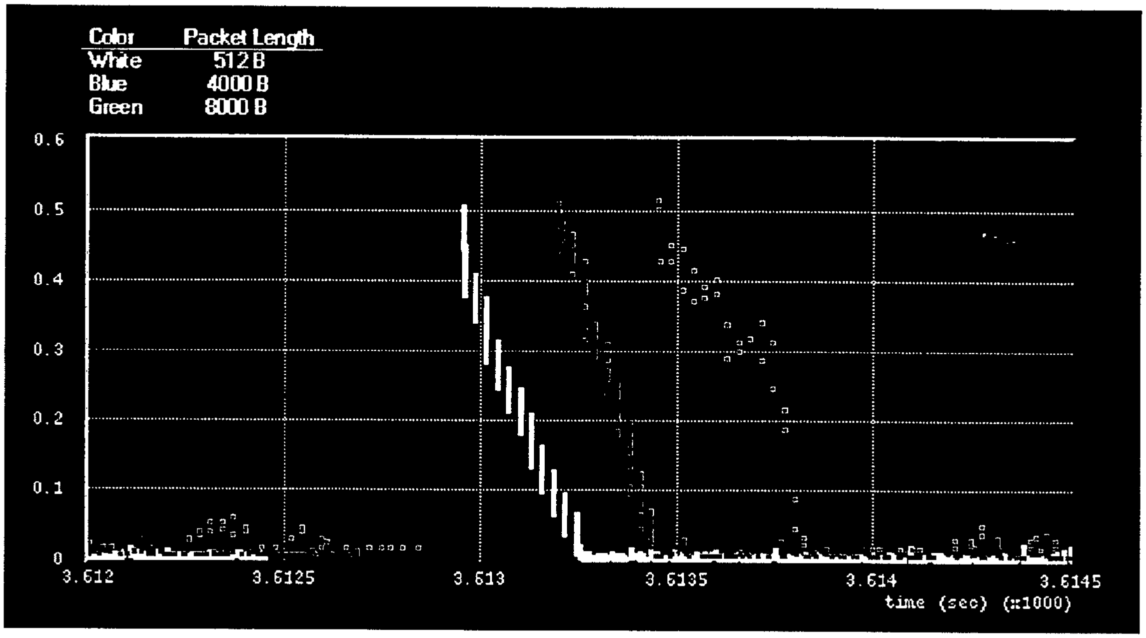


Figure 14. End-to-End Delay with BER E-7

Figure 15 show that the effect of increasing the buffer size from 65 KB to 1.2 MB has on the ETE delay when a retransmission occurs, and the case for larger buffer size (1.2 MB) and a single packet lengths (4000 B). When the first packet is discarded due to an error, the 65 KB receive buffer case retransmits the erred packet sooner than for the case of 1.2 MB receive buffer case. Both cases, however, recover at the same time since the larger receive buffer case recovers faster due to the receiver's advertised larger buffer size. After the introduction of the first error, the ETE delay for the two different simulations is no longer the same. This further affects the round trip time (RTT) parameter which is used to calculate the retransmission time out (RTO) value. Therefore, any further introduction of errors in the network will take different amounts of time to retransmit and recover as demonstrated in figure 15.

4.6. A Single Source to A Single Server with Different MAC Layer Packet Sizes

The packet sizes of the MAC layer are varied to investigate their effects on network traffic performance. The case for larger MAC layer packet sizes performs better than that of smaller packet sizes. Specifications for this case are given as follows:

Receive buffer size -	1.2 MB
ACK delays -	0.0 sec
Single Ethernet source	
Network path -	source #1 to server #1
Traffic pattern -	one way
Server type -	FDDI
Clients/server source -	4 Mbps
Application layer packet sizes -	512 B

MAC layer packet sizes in bytes - 64, 128, 536, 1000, 1456

The size of the network data packet is determined by the type of the network. For example, the maximum allowed packet size (i.e., maximum transmission unit-MTU) for the Ethernet network is 1500 B, which allows the application layer only 1456 B of data. The raw data

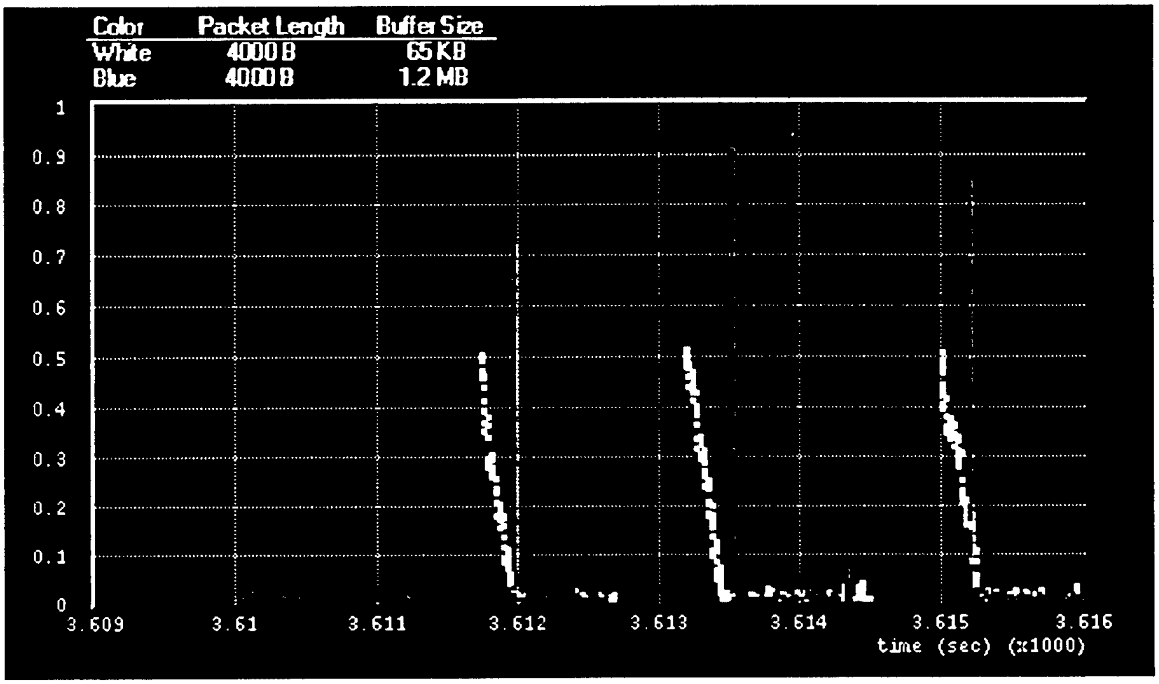


Figure 15. End-to-End Delay with BER E-7 for Different Buffer Sizes

packets smaller than 1456 B will not be fragmented while packets larger than 1456 B will be segmented and reassemble at the receiving end. The MTU packet size is varied to determine its effect on network traffic performance.

Figure 16 tests the ETE delay for MAC MTU values of 64 B and 128 B against the baseline MTU value of 1456 B. ETE delays increase for smaller MTUs by suggesting that only a partial amount of traffic is transmitted, and that a network traffic is backing up at the TCP layer. The amount of delay for the smaller MTU packet size can be substantially higher (about 10 secs in figure 16) than that of the baseline MTU. Figure 17 shows cases of larger MTU packet sizes than those of the figure 16. Average ETE delays are generally lower than 7.5 ms for the case of 536 B MTUs and MTU values closes to the baseline MTU cause even smaller ETE delays as indicated in the figure. For normal operating environments, with minimum overloading and errors, the network behaves better for larger MTU packet sizes.

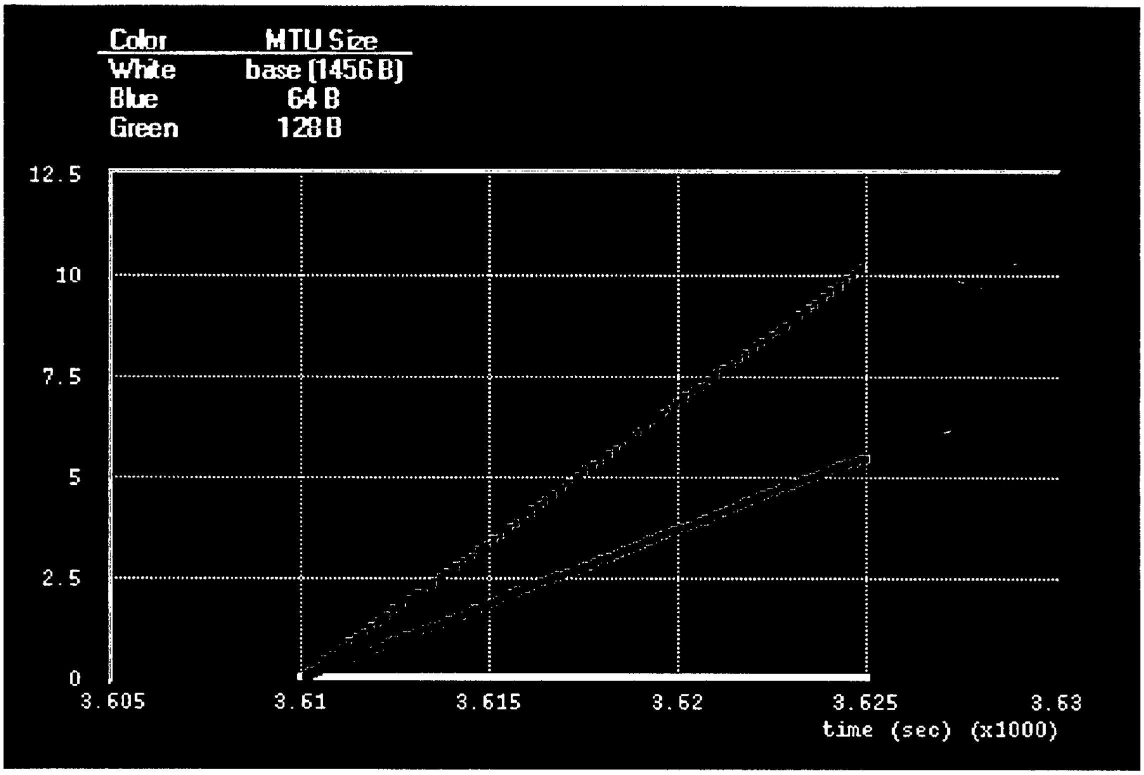


Figure 16. End-to-End Delay for Different MTU Sizes

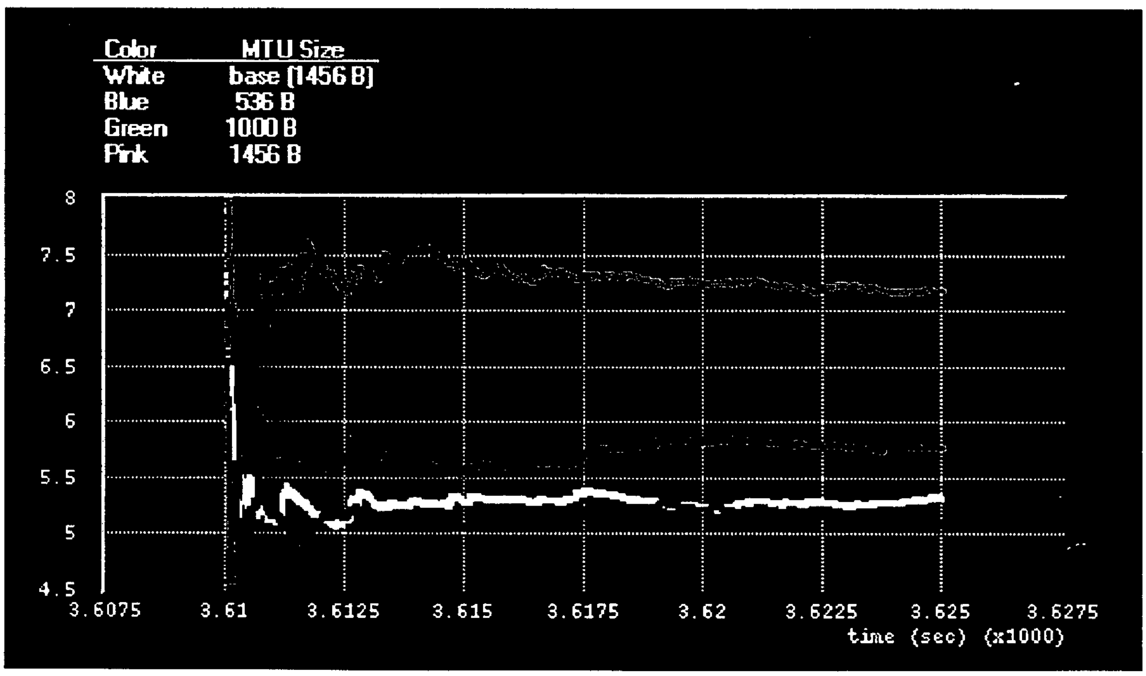


Figure 17. Average End-to-End Delay for Different MTU Sizes

Figure 18 presents simulation results for a receive buffer size of 1.2 MB, an 8 MB source rates and three MTU sizes (536 B, 1000 B, and 1456 B) to observe the network response due to a BER. For the case of a 536 B MTU, the network delay increases and does not stabilizes when the error occurs. The cases of 1000 B and 1456 B, the network recovers. This implies that a larger MTU is more efficient.

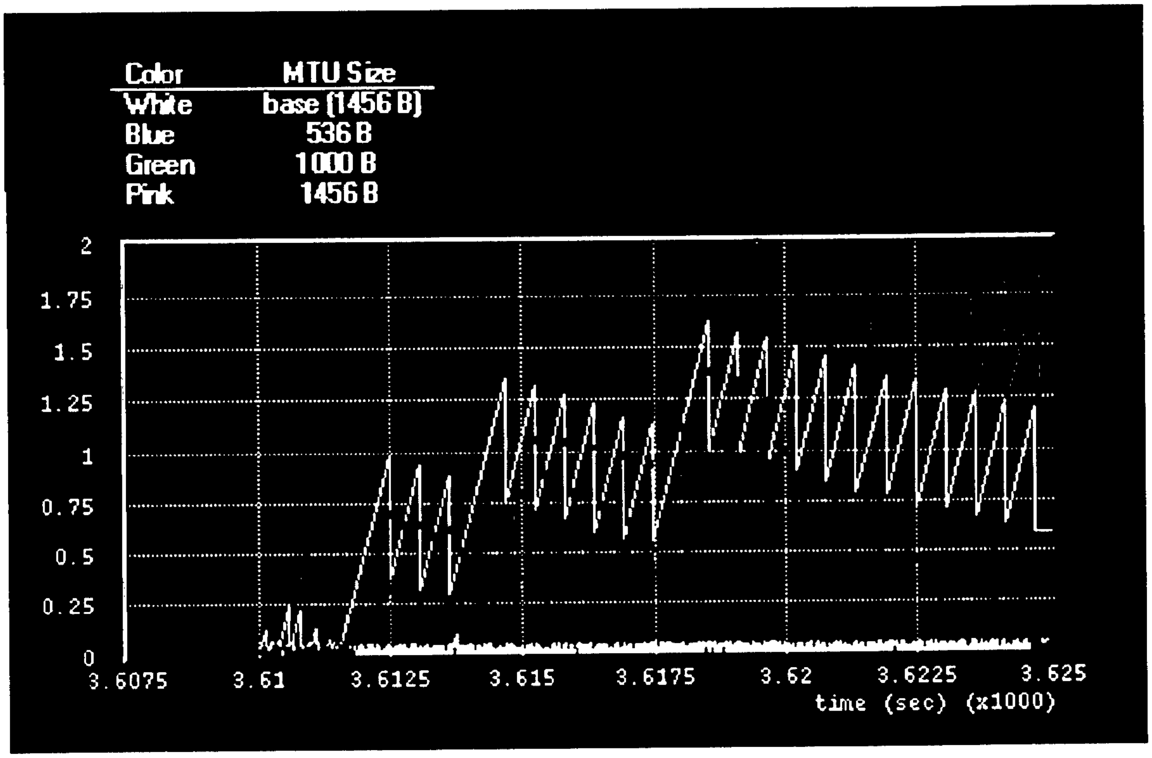


Figure 18. Average End-to-End Delay for Different MTU with BER

5. Conclusions and Recommendations

Simplified versions of the ADF backbone infrastructure are modeled, simulated, and analyzed based on parameters of packet sizes, number of nodes, source rates, BERs, receive buffer sizes, and different LAN configurations. In general, the larger MTU packets perform better even in the case of larger data rates and a single bit error. This is because more raw data application can be sent for a given amount of overhead and is therefore more efficient. For the given MTU size, we observed that the case of smaller application layer packets performed best. the application layer packets should be smaller than the MTU size in order to avoid a fragmentation. Once a fragmentation occurs, multiple MTU's are needed and therefore require more overhead for application packet. We also observed that increasing the receive buffer size can eliminate network congestion in some cases. Lessons learned from this preliminary study will enhance the ability to extend scales and complexities of the current FDDI network backbone into near real environmental network architectures and scenario. The future ATM-based backbone and other high speed network technologies will be addressed to support future mission requirements with more emphasis on the network quality of services. In general the time to send a packet depends on two factors: a propagation delay (caused by the speed of light, latencies in transmission equipment, etc) and a transmission delay that depends on the speed of the media (how many bit per seconds the medium transmits). This report emphasizes only on the propagation delay and not on the transmission delay which will be emphasized in the next report. Note that the modeling and simulation efforts in this study are limited to close a proximity between nodes without any consideration of distances between and among buildings. Future modeling and simulation efforts will be incorporated with fiber optical network architectures and related technical issues to adapt to real environmental problems. The task of network planning and management should take into account the strengths and weaknesses of operations as well as the abilities of human resources. The planners and system engineers should define, as the first step, short term goals that will advance long-range interests and also solve real operational problems. They should also prepare themselves for a great increase in the use of the networks to provide data and information to customers and consolidated missions in the future.

6. References

1. Acampora, A.S., An Introduction to Broadband Networks: LANs, MANs, ATM, B-ISDN, and Optical Network for Integrated Multimedia Telecommunications, Plenum Press, New York, NY 1994.

2. Titch, S., "Toward an Optical Layer", Telephony, vol. 233, no. 11, September 15, 1997, pp30-34.

3. Comer, D.E., Internetworking with TCP/IP: Principles, Protocols, and Architecture, 3rd Edition, vol. I, Prentice Hall, New Jersey, 1995

4. OPNET Reference Manual, MIL3, Inc., 1988, 3400 International Drive NW, Washington, DC, 20008.

5. Michael, W.H., W.J. Cronin, Jr., and K.F. Pieper, "FDDI: An Introduction to Fiber Distributed Data Interface", Digital Press, Digital Equipment Corporation, One Burlington Woods Drive, Burlington, MA 01803.

6. Zobrist, G.W., "Local Area Networks", IEEE Potentials, December '95/January '96, Issue, pp6-10, January 1996.

7. Stevens, W. R., TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley Publishing, Reading, Massachusetts, 1994